

ALIBABA CLOUD

Alibaba Cloud

资源编排
SDK参考

文档版本：20220606

 阿里云

法律声明

阿里云提醒您在阅读或使用本文档之前仔细阅读、充分理解本法律声明各条款的内容。如果您阅读或使用本文档，您的阅读或使用行为将被视为对本声明全部内容的认可。

1. 您应当通过阿里云网站或阿里云提供的其他授权通道下载、获取本文档，且仅能用于自身的合法合规的业务活动。本文档的内容视为阿里云的保密信息，您应当严格遵守保密义务；未经阿里云事先书面同意，您不得向任何第三方披露本手册内容或提供给任何第三方使用。
2. 未经阿里云事先书面许可，任何单位、公司或个人不得擅自摘抄、翻译、复制本文档内容的部分或全部，不得以任何方式或途径进行传播和宣传。
3. 由于产品版本升级、调整或其他原因，本文档内容有可能变更。阿里云保留在没有任何通知或者提示下对本文档的内容进行修改的权利，并在阿里云授权通道中不时发布更新后的用户文档。您应当实时关注用户文档的版本变更并通过阿里云授权渠道下载、获取最新版的用户文档。
4. 本文档仅作为用户使用阿里云产品及服务的参考性指引，阿里云以产品及服务的“现状”、“有缺陷”和“当前功能”的状态提供本文档。阿里云在现有技术的基础上尽最大努力提供相应的介绍及操作指引，但阿里云在此明确声明对本文档内容的准确性、完整性、适用性、可靠性等不作任何明示或暗示的保证。任何单位、公司或个人因为下载、使用或信赖本文档而发生任何差错或经济损失的，阿里云不承担任何法律责任。在任何情况下，阿里云均不对任何间接性、后果性、惩戒性、偶然性、特殊性或刑罚性的损害，包括用户使用或信赖本文档而遭受的利润损失，承担责任（即使阿里云已被告知该等损失的可能性）。
5. 阿里云网站上所有内容，包括但不限于著作、产品、图片、档案、资讯、资料、网站架构、网站画面的安排、网页设计，均由阿里云和/或其关联公司依法拥有其知识产权，包括但不限于商标权、专利权、著作权、商业秘密等。非经阿里云和/或其关联公司书面同意，任何人不得擅自使用、修改、复制、公开传播、改变、散布、发行或公开发表阿里云网站、产品程序或内容。此外，未经阿里云事先书面同意，任何人不得为了任何营销、广告、促销或其他目的使用、公布或复制阿里云的名称（包括但不限于单独为或以组合形式包含“阿里云”、“Aliyun”、“万网”等阿里云和/或其关联公司品牌，上述品牌的附属标志及图案或任何类似公司名称、商号、商标、产品或服务名称、域名、图案标示、标志、标识或通过特定描述使第三方能够识别阿里云和/或其关联公司）。
6. 如若发现本文档存在任何错误，请与阿里云取得直接联系。

通用约定

格式	说明	样例
 危险	该类警示信息将导致系统重大变更甚至故障，或者导致人身伤害等结果。	 危险 重置操作将丢失用户配置数据。
 警告	该类警示信息可能会导致系统重大变更甚至故障，或者导致人身伤害等结果。	 警告 重启操作将导致业务中断，恢复业务时间约十分钟。
 注意	用于警示信息、补充说明等，是用户必须了解的内容。	 注意 权重设置为0，该服务器不会再接受新请求。
 说明	用于补充说明、最佳实践、窍门等，不是用户必须了解的内容。	 说明 您也可以通过按Ctrl+A选中全部文件。
>	多级菜单递进。	单击设置> 网络> 设置网络类型。
粗体	表示按键、菜单、页面名称等UI元素。	在结果确认页面，单击确定。
Courier字体	命令或代码。	执行 <code>cd /d C:/window</code> 命令，进入Windows系统文件夹。
斜体	表示参数、变量。	<code>bae log list --instanceid</code> <code>Instance_ID</code>
[] 或者 [a b]	表示可选项，至多选择一个。	<code>ipconfig [-all -t]</code>
{ } 或者 {a b}	表示必选项，至多选择一个。	<code>switch {active stand}</code>

目录

1.SDK概览	05
2.Python SDK使用示例	06
3.Java SDK使用示例	12
4.Go SDK使用示例	18

1.SDK概览

资源编排服务ROS（Resource Orchestration Service）支持Java、Python、PHP、.NET、Go、Node.js和C++开发语言。

下表列举了各语言SDK的下载地址和使用说明，更多SDK的信息，请访问[阿里云开放平台](#)。

资源编排SDK下载地址	说明文档
ROS SDK for Java	开始使用
ROS SDK for Python	开始使用
ROS SDK for PHP	开始使用
ROS SDK for .NET	开始使用
ROS SDK for Go	开始使用
ROS SDK for C++	快速开始

常见问题

使用SDK时可能遇到的问题，请参见[SDK常见问题](#)。

使用示例

- Java SDK使用示例：[Java SDK使用示例](#)。
- Python SDK使用示例：[Python SDK使用示例](#)。
- 更多语言SDK使用示例：API文档提供了调试功能（需登录后使用），在任意API文档中选择调试功能，输入对应参数调用成功后可以看到详细的示例代码。

2. Python SDK使用示例

本文为您介绍资源编排服务ROS（Resource Orchestration Service）的Python SDK安装方法和示例代码。

安装Python SDK

1. 下载及安装Python SDK。

 说明 Python SDK下载和使用说明，请参见[SDK概览](#)。

2. 初始化SDK。

- i. 导入相关的软件包。

```
import sys
from typing import List
from alibabacloud_ros20190910.client import Client as ROS20190910Client
from alibabacloud_tea_openapi import models as open_api_models
from alibabacloud_ros20190910 import models as ros20190910_models
```

- ii. 初始化SDK客户端对象。

```
AK = '<yourAccessKeyId>'
SECRET = '<yourAccessKeySecret>'
REGION = '<yourRegionId>'
client = AcsClient(AK, SECRET, REGION)
```

示例

- 查询可用地域列表示例

通过调用DescribeRegions接口查询可用地域列表。关于DescribeRegions接口的详情，请参见[DescribeRegions](#)。

```
def describe_region():
    """describe regions list """
    describe_regions_request = ros20190910_models.DescribeRegionsRequest()
    response = client.describe_regions(describe_regions_request)
    return response
```

- 创建资源栈示例

通过调用CreateStack接口创建资源栈。关于CreateStack接口的详情，请参见[CreateStack](#)。

在本示例中，指定了以下参数：

- region_id：资源栈所属的地域ID。
- stack_name：资源栈名称。同一阿里云账号中资源栈名称不能重复。
- timeout_in_minutes：创建资源栈的超时时间，单位为分钟。如果在指定时间内不能完成创建，则资源栈创建失败。
- template_body：模板的主体结构。
- parameters：创建的资源栈所需要的参数。需要同时指定ParameterKey和ParameterValue。

参数设置的示例代码如下：

```
stack_name = "MyStack"
region_id = "cn-shenzhen"
timeout = 60
template_body = """
{
  "ROSTemplateFormatVersion": "2015-09-01",
  "Parameters": {
    "VpcName": {
      "Type": "String",
      "Description": "Vpc Name",
      "Label": "Vpc Name"
    },
    "CidrBlock": {
      "Type": "String",
      "Description": "Vpc CidrBlock",
      "Label": "Vpc CidrBlock"
    }
  },
  "Resources": {
    "Vpc": {
      "Type": "ALIYUN::ECS::VPC",
      "Properties": {
        "CidrBlock": {
          "Ref": "CidrBlock"
        },
        "VpcName": {
          "Ref": "VpcName"
        }
      }
    }
  }
}
"""
parameters = [
    ros20190910_models.CreateStackRequestParameters(
        parameter_key='CidrBlock',
        parameter_value='192.168.0.0/16'
    ),
    ros20190910_models.CreateStackRequestParameters(
        parameter_key='VpcName',
        parameter_value='TestVpc'
    )
]
```

创建资源栈的示例代码如下：

```
def create_stack(region_id, stack_name, template_body, timeout_in_minutes=40, parameters=
[]):
    """create stack"""
    create_stack_request = ros20190910_models.CreateStackRequest(
        region_id=region_id,
        stack_name=stack_name,
        # 若模板较大, 建议使用TemplateURL参数, 避免URL过长, 调用失败。
        # 您也可以使用Body Param + HTTP POST的方式传递模板内容。
        template_body=template_body,
        timeout_in_minutes=timeout_in_minutes,
        parameters=parameters,
        disable_rollback=True
    )
    response = client.create_stack(create_stack_request)
    return response
```

- 查询资源栈信息示例

通过调用GetStack接口查询资源栈信息。关于GetStack接口的详情, 请参见[GetStack](#)。

```
def get_stack(region_id, stack_id):
    """get descriptions of the stack"""
    get_stack_request = ros20190910_models.GetStackRequest(
        region_id=region_id,
        stack_id=stack_id,
    )
    response = client.get_stack(get_stack_request)
    return response
```

- 删除资源栈示例

通过调用DeleteStack接口删除资源栈。关于DeleteStack接口的详情, 请参见[DeleteStack](#)。

```
def delete_stack(region_id, stack_id):
    """delete stack"""
    delete_stack_request = ros20190910_models.DeleteStackRequest(
        region_id=region_id,
        stack_id=stack_id,
    )
    response = client.delete_stack(delete_stack_request)
    return response
```

- 完整示例

以下为查询可用地域列表、创建资源栈、查询资源栈和删除资源栈的完整操作示例。

```
from time import sleep
from alibabacloud_ros20190910.client import Client as ROS20190910Client
from alibabacloud_tea_openapi import models as open_api_models
from alibabacloud_ros20190910 import models as ros20190910_models
AK = '<yourAccessKeyId>'
SECRET = '<yourAccessKeySecret>'
REGION = '<yourRegionId>' # 例如: 'cn-beijing'、'cn-hangzhou'
def create_client(
    access_key_id: str,
    access_key_secret: str,
```

```

) -> ROS20190910Client:
    """
    使用AK初始化账号Client
    @param access_key_id:
    @param access_key_secret:
    @return: Client
    @throws Exception
    """
    config = open_api_models.Config(
        # 您的AccessKey ID,
        access_key_id=access_key_id,
        # 您的AccessKey Secret,
        access_key_secret=access_key_secret,
    )
    # 访问的域名
    config.endpoint = 'ros.aliyuncs.com'
    return ROS20190910Client(config)
client = create_client(AK, SECRET)
def describe_region():
    """describe regions list """
    describe_regions_request = ros20190910_models.DescribeRegionsRequest()
    response = client.describe_regions(describe_regions_request)
    return response
def create_stack(region_id, stack_name, template_body, timeout_in_minutes=40, parameters=
[]):
    """create stack"""
    create_stack_request = ros20190910_models.CreateStackRequest(
        region_id=region_id,
        stack_name=stack_name,
        # 若模板较大, 建议使用TemplateURL参数, 避免URL过长, 调用失败。
        # 您也可以使用Body Param + HTTP POST的方式传递模板内容。
        template_body=template_body,
        timeout_in_minutes=timeout_in_minutes,
        parameters=parameters,
        disable_rollback=True
    )
    response = client.create_stack(create_stack_request)
    return response.body
def get_stack(region_id, stack_id):
    """get descriptions of the stack"""
    get_stack_request = ros20190910_models.GetStackRequest(
        region_id=region_id,
        stack_id=stack_id,
    )
    response = client.get_stack(get_stack_request)
    return response.body
def delete_stack(region_id, stack_id):
    """delete stack"""
    delete_stack_request = ros20190910_models.DeleteStackRequest(
        region_id=region_id,
        stack_id=stack_id,
    )
    response = client.delete_stack(delete_stack_request)
    return response.body

```

```

if __name__ == '__main__':
    test_template = """
    {
        "ROSTemplateFormatVersion": "2015-09-01",
        "Parameters": {
            "VpcName": {
                "Type": "String",
                "Description": "Vpc Name",
                "Label": "Vpc Name"
            },
            "CidrBlock": {
                "Type": "String",
                "Description": "Vpc CidrBlock",
                "Label": "Vpc CidrBlock"
            }
        },
        "Resources": {
            "Vpc": {
                "Type": "ALIYUN::ECS::VPC",
                "Properties": {
                    "CidrBlock": {
                        "Ref": "CidrBlock"
                    },
                    "VpcName": {
                        "Ref": "VpcName"
                    }
                }
            }
        }
    }
    """

    parameters = [
        ros20190910_models.CreateStackRequestParameters(
            parameter_key='CidrBlock',
            parameter_value='192.168.0.0/16'
        ),
        ros20190910_models.CreateStackRequestParameters(
            parameter_key='VpcName',
            parameter_value='TestVpc'
        )
    ]

    region_id = REGION
    describe_region()
    stack = create_stack(region_id, 'MyStack', test_template, parameters=parameters, time
out_in_minutes=60)
    stack_status = None
    while True:
        stack_info = get_stack(region_id, stack.stack_id)
        stack_status = stack_info.status
        print('Stack Id: {}, Stack状态: {}'.format(stack.stack_id, stack_status))
        # 等待创建资源栈任务执行完毕。
        if stack_status == 'CREATE_COMPLETE':
            break
        sleep(3)
    delete_stack(region_id, stack.stack_id)

```

```
delete_stack(region_id, stack.stack_id)
```

3.Java SDK使用示例

本文为您介绍资源编排服务ROS（Resource Orchestration Service）的Java SDK安装方法和示例代码。

安装Java SDK

1. 下载及安装Java SDK。

? 说明

- Java SDK的下载和使用说明，请参见[SDK概览](#)。
- 建议您使用JRE 1.8及以上版本。

2. 在pom.xml中，添加依赖包。

```
<dependencies>
  <dependency>
    <groupId>com.aliyun</groupId>
    <artifactId>tea-openapi</artifactId>
    <version>0.0.19</version>
  </dependency>
  <dependency>
    <groupId>com.aliyun</groupId>
    <artifactId>ros20190910</artifactId>
    <version>1.0.1</version>
  </dependency>
</dependencies>
```

3. 初始化SDK。

- i. 导入相关的软件包。

```
import com.aliyun.tea.*;
import com.aliyun.ros20190910.*;
import com.aliyun.ros20190910.models.*;
import com.aliyun.teaopenapi.*;
import com.aliyun.teaopenapi.models.*;
```

- ii. 初始化SDK客户端对象。

```
public static com.aliyun.ros20190910.Client createClient(String accessKeyId, String
accessKeySecret) throws Exception {
    Config config = new Config()
        // 您的AccessKey ID
        .setAccessKeyId(accessKeyId)
        // 您的AccessKey Secret
        .setAccessKeySecret(accessKeySecret);
    // 访问的域名
    config.endpoint = "ros.aliyuncs.com";
    return new com.aliyun.ros20190910.Client(config);
}
```

示例

- 查询可用地域列表示例

通过调用DescribeRegions接口查询可用地域列表。关于DescribeRegions接口的详情，请参见[DescribeRegions](#)。

```
public static void describeRegions(String[] args_) throws Exception {
    java.util.List<String> args = java.util.Arrays.asList(args_);
    com.aliyun.ros20190910.Client client = Sample.createClient("accessKeyId", "access
KeySecret");
    DescribeRegionsRequest describeRegionsRequest = new DescribeRegionsRequest();
    DescribeRegionsResponse resp = client.describeRegions(describeRegionsRequest);
}
```

- 创建资源栈示例

通过调用CreateStack接口创建资源栈。关于CreateStack接口的详情，请参见[CreateStack](#)。

```

public static void createStack(String stackName, String regionId) throws Exception {
    java.util.List<String> args = java.util.Arrays.asList(args_);
    CreateStackRequest.CreateStackRequestParameters parameters0 = new CreateStackRequest.
CreateStackRequestParameters()
        .setParameterKey("VpcName")
        .setParameterValue("TestVpc");
    CreateStackRequest.CreateStackRequestParameters parameters1 = new CreateStackRequest.
CreateStackRequestParameters()
        .setParameterKey("CidrBlock")
        .setParameterValue("192.168.0.0/16");
    CreateStackRequest createStackRequest = new CreateStackRequest()
        .setTemplateBody("{\n" +
            "  \"ROSTemplateFormatVersion\": \"2015-09-01\",\n" +
            "  \"Parameters\": {\n" +
            "    \"VpcName\": {\n" +
            "      \"Type\": \"String\",\n" +
            "      \"Description\": \"Vpc Name\",\n" +
            "      \"Label\": \"Vpc Name\"\n" +
            "    },\n" +
            "    \"CidrBlock\": {\n" +
            "      \"Type\": \"String\",\n" +
            "      \"Description\": \"Vpc CidrBlock\",\n" +
            "      \"Label\": \"Vpc CidrBlock\"\n" +
            "    }\n" +
            "  },\n" +
            "  \"Resources\": {\n" +
            "    \"Vpc\": {\n" +
            "      \"Type\": \"ALIYUN::ECS::VPC\",\n" +
            "      \"Properties\": {\n" +
            "        \"CidrBlock\": {\n" +
            "          \"Ref\": \"CidrBlock\"\n" +
            "        },\n" +
            "        \"VpcName\": {\n" +
            "          \"Ref\": \"VpcName\"\n" +
            "        }\n" +
            "      }\n" +
            "    }\n" +
            "  }\n" +
            "}")
        .setStackName("MyStack")
        .setRegionId("cn-hangzhou")
        .setTimeoutInMinutes(40L)
        .setParameters(java.util.Arrays.asList(
            parameters0,
            parameters1
        ));
    // 复制代码运行后, 请自行打印API的返回值。
    client.createStack(createStackRequest);
}

```

- 查询资源栈示例

通过调用GetStack接口查询资源栈信息。关于GetStack接口的详情, 请参见[GetStack](#)。

```
public static void getStack(String stackId, String regionId) throws Exception {
    GetStackRequest getStackRequest = new GetStackRequest()
        .setStackId(stackId)
        .setRegionId(regionId);
    // 复制代码运行后，请自行打印API的返回值。
    GetStackResponse resp = client.getStack(getStackRequest);
    com.aliyun.teaconsole.Client.log(com.aliyun.teautil.Common.toJSONString(TeaModel.buildMap(resp)));
}
```

- 删除资源栈示例

通过调用DeleteStack接口删除资源栈。关于DeleteStack接口的详情，请参见[DeleteStack](#)。

```
public static void deleteStack(String stackId, String regionId) throws Exception {
    DeleteStackRequest deleteStackRequest = new DeleteStackRequest()
        .setStackId(stackId)
        .setRegionId(regionId);
    // 复制代码运行后，请自行打印API的返回值。
    DeleteStackResponse resp = client.deleteStack(deleteStackRequest);
    com.aliyun.teaconsole.Client.log(com.aliyun.teautil.Common.toJSONString(TeaModel.buildMap(resp)));
}
```

- 完整示例

以下为查询可用地域列表、创建资源栈、查询资源栈和删除资源栈的完整操作示例。

```
package com.aliyun.sample;
import com.aliyun.ros20190910.Client;
import com.aliyun.tea.*;
import com.aliyun.ros20190910.*;
import com.aliyun.ros20190910.models.*;
import com.aliyun.teaopenapi.*;
import com.aliyun.teaopenapi.models.*;
public class Sample {
    /**
     * 使用AK和SK初始化账号Client
     *
     * @param accessKeyId
     * @param accessKeySecret
     * @return Client
     * @throws Exception
     */
    private static String ACCESSKEYID = "<yourAccessKeyId>";
    private static String SECRET = "<yourAccessKeySecret>";
    private static String REGIONID = "<yourRegionId>";
    public static com.aliyun.ros20190910.Client createClient() throws Exception {
        Config config = new Config()
            // 您的AccessKey ID
            .setAccessKeyId(ACCESSKEYID)
            // 您的AccessKey Secret
            .setAccessKeySecret(SECRET);
        // 访问的域名
        config.endpoint = "ros.aliyuncs.com";
        return new com.aliyun.ros20190910.Client(config);
    }
}
```

```

        return new com.aliyun.ros20190910.Client(config);
    }

    public static void describeRegions(Client client) throws Exception {
        DescribeRegionsRequest describeRegionsRequest = new DescribeRegionsRequest();
        // 复制代码运行后，请自行打印API的返回值。
        DescribeRegionsResponse resp = client.describeRegions(describeRegionsRequest);
    }

    public static void getStack(Client client, String stackId, String regionId) throws Exception {
        GetStackRequest getStackRequest = new GetStackRequest()
            .setStackId(stackId)
            .setRegionId(regionId);
        // 复制代码运行后，请自行打印API的返回值。
        GetStackResponse resp = client.getStack(getStackRequest);
        com.aliyun.teaconsole.Client.log(com.aliyun.teautil.Common.toJSONString(TeaModel.buildMap(resp)));
    }

    public static void deleteStack(Client client, String stackId, String regionId) throws Exception {
        DeleteStackRequest deleteStackRequest = new DeleteStackRequest()
            .setStackId(stackId)
            .setRegionId(regionId);
        // 复制代码运行后，请自行打印API的返回值。
        DeleteStackResponse resp = client.deleteStack(deleteStackRequest);
        com.aliyun.teaconsole.Client.log(com.aliyun.teautil.Common.toJSONString(TeaModel.buildMap(resp)));
    }

    public static void main(String[] args_) throws Exception {
        java.util.List<String> args = java.util.Arrays.asList(args_);
        com.aliyun.ros20190910.Client client = Sample.createClient();
        CreateStackRequest.CreateStackRequestParameters parameters0 = new CreateStackRequest.CreateStackRequestParameters()
            .setParameterKey("VpcName")
            .setParameterValue("TestVpc");
        CreateStackRequest.CreateStackRequestParameters parameters1 = new CreateStackRequest.CreateStackRequestParameters()
            .setParameterKey("CidrBlock")
            .setParameterValue("192.168.0.0/16");
        CreateStackRequest createStackRequest = new CreateStackRequest()
            .setTemplateBody("{\n" +
                "  \"ROSTemplateFormatVersion\": \"2015-09-01\",\n" +
                "  \"Parameters\": {\n" +
                "    \"VpcName\": {\n" +
                "      \"Type\": \"String\",\n" +
                "      \"Description\": \"Vpc Name\",\n" +
                "      \"Label\": \"Vpc Name\"\n" +
                "    },\n" +
                "    \"CidrBlock\": {\n" +
                "      \"Type\": \"String\",\n" +
                "      \"Description\": \"Vpc CidrBlock\",\n" +
                "      \"Label\": \"Vpc CidrBlock\"\n" +
                "    }\n" +
                "  },\n" +
                "  \"Resources\": {\n" +
                "    \"Vpc\": {\n"
            );
    }

```

```
        "Type": "ALIYUN::ECS::VPC", \n" +
        "Properties": {\n" +
        "  CidrBlock": {\n" +
        "    Ref": "CidrBlock"\n" +
        "  }, \n" +
        "  VpcName": {\n" +
        "    Ref": "VpcName"\n" +
        "  } \n" +
        " } \n" +
        " } \n" +
        " }")

    .setStackName("stackName")
    .setRegionId("cn-hangzhou")
    .setTimeoutInMinutes(40L)
    .setParameters(java.util.Arrays.asList(
        parameters0,
        parameters1
    ));

// 复制代码运行后，请自行打印API的返回值。
client.createStack(createStackRequest);
Sample.describeRegions(client);
Sample.getStack(client, "yourStackId", REGIONID);
Sample.deleteStack(client, "yourStackId", REGIONID);
}
}
```

4.Go SDK使用示例

本文为您介绍资源编排服务ROS（Resource Orchestration Service）的Go SDK安装方法和示例代码。

安装Go语言

1. 下载及安装Go语言。

说明

- Go语言的下载及安装，请参见[官方安装文档](#)。
- 建议您使用Go语言1.10.1及以上版本。

2. 验证安装是否成功。

- 验证Go语言版本。

执行以下命令：

```
go version
```

预期输出：

```
go version go1.10.1
```

- 验证GOPATH配置。

执行以下命令：

```
go env GOPATH
```

预期输出：

```
/root/gos
```

安装Go SDK

1. 下载及安装Go SDK。

说明 Go SDK的下载和使用说明，请参见[SDK概览](#)。

2. 初始化SDK。

- i. 导入相关的软件包。

```
import (  
    "os"  
    ros20190910 "github.com/alibabacloud-go/ros-20190910/v3/client"  
    openapi "github.com/alibabacloud-go/darabonba-openapi/client"  
    util "github.com/alibabacloud-go/tea-utils/service"  
    "github.com/alibabacloud-go/tea/tea"  
)
```

ii. 初始化SDK客户端对象。

```
func CreateClient (accessKeyId *string, accessKeySecret *string) (_result *ros20190910.Client, _err error) {
    config := &openapi.Config{
        // 您的AccessKey ID
        AccessKeyId: accessKeyId,
        // 您的AccessKey Secret
        AccessKeySecret: accessKeySecret,
    }
    // 访问的域名
    config.Endpoint = tea.String("ros.aliyuncs.com")
    _result = &ros20190910.Client{}
    _result, _err = ros20190910.NewClient(config)
    return _result, _err
}
```

示例

● 查询可用地域列表示例

通过调用DescribeRegions接口查询可用地域列表。关于DescribeRegions接口的详情，请参见[DescribeRegions](#)。

```
func _main (args []*string) (_err error) {
    client, _err := CreateClient(tea.String("accessKeyId"), tea.String("accessKeySecret"))
    if _err != nil {
        return _err
    }
    describeRegionsRequest := &ros20190910.DescribeRegionsRequest{}
    runtime := &util.RuntimeOptions{}
    // 复制代码运行请自行打印API的返回值
    _, _err = client.DescribeRegionsWithOptions(describeRegionsRequest, runtime)
    if _err != nil {
        return _err
    }
    return _err
}
```

● 创建资源栈示例

通过调用CreateStack接口创建资源栈。关于CreateStack接口的详情，请参见[CreateStack](#)。

在本示例中，指定了以下参数：

- StackName：资源栈名称。同一阿里云账号中资源栈名称不能重复。
- TemplateBody：模板的主体结构。
- Parameter：创建资源栈所需要的参数。需要同时指定ParameterKey和ParameterValue。

```
func CreateStackSample (client *ros20190910.Client, regionId *string, stackName *string,
templateBody *string, parameter []*string) (_result *string, _err error) {
    request := &ros20190910.CreateStackRequest{
        RegionId: regionId,
        StackName: stackName,
        TemplateBody: templateBody,
        Parameters: []*ros20190910.CreateStackRequestParameters{parameter},
    }
    response, _err := client.CreateStack(request)
    if _err != nil {
        return _result, _err
    }
    _result = response.Body.StackId
    return _result , _err
}
```

- 查询资源栈信息示例

通过调用GetStack接口查询资源栈信息。关于GetStack接口的详情，请参见[GetStack](#)。

```
func GetStackSample (client *ros20190910.Client, regionId *string, stackId *string) (_res
ult *string, _err error) {
    request := &ros20190910.GetStackRequest{
        RegionId: regionId,
        stackId: stackId,
    }
    response, _err := client.GetStack(request)
    if _err != nil {
        return _result, _err
    }
    _result = response.Body.StackId
    return _result , _err
}
```

- 删除资源栈示例

通过调用DeleteStack接口删除资源栈。关于DeleteStack接口的详情，请参见[DeleteStack](#)。

```
func DeleteStackSample (client *ros20190910.Client, regionId *string, stackId *string) (_
err error) {
    request := &ros20190910.DeleteStackRequest{
        RegionId: regionId,
        stackId: stackId,
    }
    _, _err := client.DeleteStack(request)
    if _err != nil {
        return _err
    }
    _result = response.Body.StackId
    return _err
}
```

- 完整示例

以下为查询可用地域列表、创建资源栈、查询资源栈和删除资源栈的完整操作示例。

```

package main
import (
    "fmt"
    "os"
    openapi "github.com/alibabacloud-go/darabonba-openapi/client"
    ros20190910 "github.com/alibabacloud-go/ros-20190910/v2/client"
    console "github.com/alibabacloud-go/tea-console/client"
    util "github.com/alibabacloud-go/tea-utils/service"
    "github.com/alibabacloud-go/tea/tea"
)
/**
 * 使用AccessKey ID和AccessKey Secret初始化客户端对象
 * @param accessKeyId
 * @param accessKeySecret
 * @param regionId
 * @return Client
 * @throws Exception
 */
func Initialization(accessKeyId *string, accessKeySecret *string, regionId *string) (_result *ros20190910.Client, _err error) {
    config := &openapi.Config{}
    // 您的AccessKey ID
    config.AccessKeyId = accessKeyId
    // 您的AccessKey Secret
    config.AccessKeySecret = accessKeySecret
    // 您的地域ID
    config.RegionId = regionId
    _result = &ros20190910.Client{}
    _result, _err = ros20190910.NewClient(config)
    return _result, _err
}
/**
 * 创建资源栈
 * @param client
 * @param request
 * @throws Exception
 */
func CreateStackSample(client *ros20190910.Client, regionId *string, stackName *string, templateBody *string, parameter *string) (_result *string, _err error) {
    paramMap := util.AssertAsMap(util.ParseJSON(parameter))
    param := &ros20190910.CreateStackRequestParameters{
        ParameterKey: util.AssertAsString(paramMap["ParameterKey"]),
        ParameterValue: util.AssertAsString(paramMap["ParameterValue"]),
    }
    request := &ros20190910.CreateStackRequest{
        RegionId: regionId,
        StackName: stackName,
        TemplateBody: templateBody,
        Parameters: []*ros20190910.CreateStackRequestParameters{param},
    }
    response, _err := client.CreateStack(request)
    if _err != nil {
        return _result, _err
    }
    console.Log(tea.String("创建中 资源栈ID: ") + tea.StringValue(response.Body.StackId))
}

```

```

        console.Log(tea.String("创建中，资源栈ID:" + tea.StringValue(response.Body.StackId)))
        fmt.Print(tea.String("创建中，资源栈ID:" + tea.StringValue(response.Body.StackId)))
        _result = response.Body.StackId
        return _result, _err
    }
}
/**
 * 查询资源栈信息
 * @param client
 * @param stackId
 * @throws Exception
 */
func GetStackSample(client *ros20190910.Client, regionId *string, stackId *string, wait *
bool) (_err error) {
    request := &ros20190910.GetStackRequest{
        RegionId: regionId,
        StackId:  stackId,
    }
    response, _err := client.GetStack(request)
    if _err != nil {
        return _err
    }
    body := response.Body
    fmt.Print(util.ToJSONString(tea.ToMap(response)))
    if tea.BoolValue(wait) {
        if tea.BoolValue(util.EqualString(body.Status, tea.String("CREATE_IN_PROGRESS")))
        {
            fmt.Print(tea.String("正在创建中....."))
            _err = util.Sleep(tea.Int(30000))
            if _err != nil {
                return _err
            }
            _err = GetStackSample(client, regionId, stackId, wait)
            if _err != nil {
                return _err
            }
        } else if tea.BoolValue(util.EqualString(body.Status, tea.String("CREATE_FAILED")
        )) {
            fmt.Print(tea.String("创建失败，信息:" + tea.StringValue(body.StatusReason) + "
, requestId:" + tea.StringValue(body.RequestId)))
        } else if tea.BoolValue(util.EqualString(body.Status, tea.String("CREATE_COMPLETE
        "))) {
            fmt.Print(tea.String("创建成功"))
            fmt.Print(tea.String("ID:" + tea.StringValue(body.StackId)))
        }
    } else {
        fmt.Print(tea.String("ID:" + tea.StringValue(body.StackId)))
        fmt.Print(tea.String("状态:" + tea.StringValue(body.Status)))
    }
    return _err
}
func DeleteStackSample(client *ros20190910.Client, regionId *string, stackId *string) (_e
rr error) {
    request := &ros20190910.DeleteStackRequest{
        RegionId: regionId,
        StackId:  stackId.

```

```

    }
    _, _err = client.DeleteStack(request)
    if _err != nil {
        return _err
    }
    return _err
}

func _main(args []*string) (_err error) {
    // access key id
    accessKeyId := tea.String(`You access key id`)
    // 区域ID
    accessKeySecret := tea.String(`you access key secret`)
    // 区域ID
    regionId := tea.String(`cn-hangzhou`)
    // 资源栈名称
    stackName := tea.String(`MyTestStack`)
    // 创建模板
    templateBody := tea.String(`
        {
        "ROSTemplateFormatVersion": "2015-09-01",
        "Parameters": {
            "CidrBlock": {
                "Type": "String",
                "Description": "Vpc CidrBlock",
                "Label": "Vpc CidrBlock"
            }
        },
        "Resources": {
            "Vpc": {
                "Type": "ALIYUN::ECS::VPC",
                "Properties": {
                    "CidrBlock": {
                        "Ref": "CidrBlock"
                    },
                    "VpcName": {
                        "Fn::Join": [
                            "-",
                            [
                                "StackId",
                                {
                                    "Ref": "ALIYUN::StackId"
                                }
                            ]
                        ]
                    }
                }
            }
        }
        }`)
    // 模板参数 (JSON)
    parameter := tea.String(`{"ParameterKey": "CidrBlock", "ParameterValue": "192.168.XX.XX/16"}`)
    // 等待创建成功
    wait := tea.Bool(false)

```

```
client, _err := Initialization(accessKeyId, accessKeySecret, regionId)
if _err != nil {
    return _err
}
fmt.Print(tea.String("-----创建资源栈-----"))
stackId, _err := CreateStackSample(client, regionId, stackName, templateBody, parameter)
if _err != nil {
    return _err
}
_err = util.Sleep(tea.Int(1000))
if _err != nil {
    return _err
}
fmt.Print(tea.String("-----资源栈" + tea.StringValue(stackName) + "信息-----"))
_err = GetStackSample(client, regionId, stackId, wait)
if _err != nil {
    return _err
}
return _err
fmt.Print(tea.String("-----删除资源栈-----"))
_err = DeleteStackSample(client, regionId, stackId)
if _err != nil {
    return _err
}
return _err
}
func main() {
    err := _main(tea.StringSlice(os.Args[1:]))
    if err != nil {
        panic(err)
    }
}
```